

FTPMAN Log

Request activity

Tue, Oct 15, 2002

There already exists a log for RETDAT requests, where the system code logs arrival of RETDAT requests that include something from the local node, plus the cancelation of such requests. This notes discusses how we might add a log for FTPMAN requests.

How does one characterize a FTPMAN request? For the case of a RETDAT request, we only saved the requesting node, the number of bytes of data sought by the request, the number of devices included in the request, and the Frequency Time Descriptor (FTD). For an FTPMAN request, we need to know the type of request (continuous or snapshot). For the continuous case, we could be interested in the data sample period, the reply period, the number of devices, the number of bytes requested, and whether it is regarded as a server request. For the snapshot case, we could be interested in the data sample period (or the sample frequency in Hz), the delay, the arm clock event, and the number of data points to be collected. Putting this in a list, we have:

Continuous

sample frequency in Hz (2)
number of devices (1)
number bytes requested (2)
server request? (hi bit of #devices?)
reply period (1)

Snapshot

sample frequency in Hz (4)
number of devices (1)
number bytes requested (2)
server request? (hi bit of #devices?)

It seems that this may almost fit within 8 bytes, but we must also have the requesting node# and the message-id used. This is so the cancel record can be matched to a request. For the RETDAT case, the message-id usurped the yr-mo part of the time-of-day, which was ok. The same trick could be used here.

Continuous

rqNode	2	requesting node#
nBytes	2	#bytes requested
nDev	1	#devices
rpyPer	1	reply period
sampFreq	2	sample frequency
msgId	2	message-id from Acnet header
daHrMnScCyMs	6	time-of-day

Snapshot

rqNode	2	requesting node#
nBytes	2	#bytes requested
nDev	1	#devices in upper 4 bits
sampFreq	3	sample frequency (28 bits)
msgId	2	message-id from Acnet header
daHrMnScCyMs	6	time-of-day

One can tell whether it is a continuous or snapshot record by the 5th byte. For the continuous case, it is a hex value between 01–05; for the snapshot case, it is a value with the upper 4 bits set, so that the byte value is a hex value in the range 1x–4x. (Goofy, huh?) But we still need the server flag.

But suppose we are willing to allocate 32 bytes per record. Consider the following:

<i>Field</i>	<i>Size</i>	<i>Meaning</i>
rqNode	2	requesting node#
msgId	2	message-id from Acnet header
nBytes	2	#bytes requested
rpyPer	1	reply period, if continuous
typeCode	1	FTPMAN type code
nDev	1	#devices in upper 4 bits
error	1	FTPMAN error code
delay	2	delay, if snapshot
sampFreq	4	sample frequency in Hz
ssdn	8	First SSDN
time	8	time-of-day